An Adaptive Pitch Control Strategy for a Doubly Fed Wind Generation System

Syed Ahmed Raza∗ & A.H.M.A. Rahim†
∗Department of Electrical Engineering, Prince Mohammad Bin Fahd University
Al-Khobar, Saudi Arabia,∗email: snaqvi@pmu.edu.sa
†Department of Electrical Engineering, King Fahd University of Petroleum and Minerals
Dhahran, Saudi Arabia,†email: ahrahim@kfupm.edu.sa

Keywords: Doubly fed induction generator, Pitch control, Adaptive neural network, Differential evolution

Abstract
A smart pitch control strategy for a variable speed doubly fed wind generation system is presented in this article. Nonlinear as well as linearized dynamic models of the wind system pitch controller and the doubly fed induction generator including the drive train are developed. A PI controller is employed to generate the appropriate pitch angle for varying wind speed conditions. An adaptive artificial neural network (ANN) is trained to produce PI gain settings for various wind speed conditions. The training data, on the other hand, was generated through differential evolution (DE). Simulation studies show that the DE based adaptive ANN can generate the appropriate control to deliver the wind power to the generator efficiently with minimum transients. The data used was collected from the wind generator located at the King Fahd University beach front.

1. INTRODUCTION
Renewable energy is gaining momentum especially because of the diminishing oil reserves and environmental concerns. Wind energy in itself encompasses various engineering fields and is rapidly developing into a multi-disciplinary area of research and experimentation [1]. In wind generation systems, the variable speed doubly fed induction generator (DFIG) is preferred over the permanent magnet synchronous generator (PMSG) type for its versatility [2].

Wind turbines, in general, have two controls. The mechanical control which includes the blade pitch control and the electrical control covers the control of the power converters and the load side control. The control of blade pitch angle is a necessary part of variable speed wind turbines as by controlling the pitch angle, we can control the aerodynamic power that flows through to the generator. By carefully controlling the pitch angle, we can allow for maximum power being delivered to the generator at a particular wind speed to which the turbine is exposed [3].

The control of blade pitch angle is a necessary part of variable speed wind turbines since by controlling the pitch angle, the aerodynamic power that flows through to the generator can be adjusted.

The system dynamics using the PI controller have been described in [5]-[7]. Fuzzy logic was used in [8] to find the pitch controller parameters. A self tuning fuzzy PID controller was proposed in [9]. Robust controllers for output power leveling of variable speed variable pitch wind turbine generator systems are also available in literature. The use of generalized predictive control has been reported in [10] and [11]. A neural network capable of self tuning for during different operating conditions has been reported in [12]. The proposed controller consists of neural networks inverse and forward identifiers for modeling the dynamics of the system. One of the challenging aspects of wind generation system is to extract maximum power from randomly changing wind conditions. At the same time the turbine should have higher efficiency to transfer maximum power to the grid. In this work, a variable speed DFIG system will be investigated for pitch control. Pitch controller parameters are to be found using an adaptive intelligent control for maximum power transfer from wind.

The organization of the article is as follows: Section II includes a complete dynamic model of DFIG equipped with pitch controller, Section III describes the adaptive pitch control strategy making use of the back propagation algorithm and differential evolution, Section IV discusses the simulations carried out with the proposed control strategy and at the end, the results are summarized in the conclusion.

2. SYSTEM MODELING
A schematic diagram of the DFIG system connected to the power grid equipped with pitch control is shown in figure1. The induction generator is driven by a horizontal axis wind turbine connected through its gear boxes. The converters are located between the rotor terminals and the grid. The dynamic model of the system includes the wind turbine, pitch controller and the generator with its converters.

2.1. Wind Turbine Aerodynamics
The amount of power extracted from wind is a function of air density and is given by [2],

\[ P_m = \frac{1}{2} \rho \pi R^2 V_w^3 C_p(\lambda, \beta) \]  

where, \( V_w \) is the wind velocity, \( R \) is the radius of the rotor blades and \( C_p \) is the power coefficient that is dependent upon
the tip speed ratio \( \lambda \) and the pitch angle \( \beta \). The power coefficient \( C_p \) is a non-linear function of \( \lambda \) and \( \beta \) given as,

\[
C_p(\lambda, \beta) = 0.5176 \left[ \frac{116}{\lambda} - 0.4 \beta - 5 \right] e^{-\frac{21}{\lambda}} + 0.0068 \lambda
\]

\[
\frac{1}{\lambda} = \frac{1}{\lambda + 0.08 \beta} - \frac{0.035}{\beta^3 + 1}
\]

(2)

where, \( \omega_0 \) is the turbine speed. The variation of mechanical power with respect to the rotor speed for a particular wind speed for different values of pitch angle is given in Figure 2.

### 2.2. Induction Generator Model

In dynamic modeling, the DFIG is normally represented by a 4th order model of stator and rotor currents along the d-q axes and are given as. These equations are dependent on generator slip. The converter dynamic model is normally represented by a second order model containing d-q components of converter currents in addition to the converter DC capacitor voltage \( V_C \). The drive train model consists of the high inertia turbine coupled to relatively lower inertia generator and are expressed in terms of a third order model with states \( \omega_r, \omega_c, \theta_s \) which are turbine and generator speeds, and torsion angle, respectively.

### 2.3. Pitch angle control

As can be observed from (1) and (2), control of pitch angle \( \beta \) provides an effective means for controlling the power input to the generator under varying wind speeds. Pitch servos provide the necessary angle to the blades. This may be hydraulic or electrical systems. Conventional pitch angle control use PI controllers to generate the appropriate \( \beta \). The pitch servo system compares the measured angle of \( \beta \) to the reference and corrects the error. Usually a first order servo model is sufficient in investigations of power system stability. However, more detailed pitch servo models may be used also.

In order to get more realistic response of the generic, regular pitch control a number of delay mechanisms must be implemented in control system models. Such delays represent sampling and filters, damping natural frequencies in wind turbine construction.

A popular method of employing the pitch control is to make use of the generator power as a feedback signal compared with the reference mechanical power available from wind. Here \( P_g \) represents the electrical power output from the system. The controller equations can be derived as,

where, \( K_P \) and \( K_I \) are the controller gains, \( \beta_I \) is an intermediate state, \( \Delta P \) is the error between the reference power and the generated power while \( \beta \) is the actual pitch setting from the controller.

### 2.4. Non-Linear Model

Combining the dynamics of generator, drive train, converter circuits and the pitch controller, the composite dynamic model can be written as,

\[
\dot{x} = f(x, u)
\]

Here, \( x \) is the vector of the states consisting of \([i_{ds}, i_{qr}, i_{dr}, i_{qr}, \omega_r, \omega_c, \theta_s, \omega_p, \theta_r, i_{ds}, i_{qr}, V_C, \beta_I] \) and \( u \) is the pitch control. In this work the gains of the PI controller are obtained from a trained artificial neural network (ANN).

### 2.5. Linearized Model

The linearized of (3) can be expressed as,

\[
\dot{x} = Ax + Bu
\]

\[
y = Cy + Du
\]

(4)

Where, the states are the perturbations of the original state variables in (3). Note the linearization process requires finding \( \Delta P_m \), which can be written as,

\[
\Delta P_m = K_i \Delta \omega_r + K_\beta \Delta \beta
\]

(5)

where \( K_i = \left. \frac{\partial P_m}{\partial \omega_r} \right|_{op} \) and \( K_\beta = \left. \frac{\partial P_m}{\partial \beta} \right|_{op} \)

The wind data for the training was collected from the 5-kW wind system installed at the King Fahd University beach front. The training data was generated by a differential evolution technique. A brief outline of the ANN and DE procedures are given in the following.

### 3. ADAPTIVE PITCH CONTROL STRATEGY

Figure 4 shows the adaptive pitch control strategy employed for DFIG. The strategy involves making use of input wind speed and setting the reference power by making use of figure 2. The reference neural network model uses a back propagation based neural network which sets the PI gains as targets with random wind speeds as inputs. Using differential evolution, the training data for this neural network is obtained. The PI gains obtained from DE are optimal gains by making use of eigenvalue based objective function.

As the wind speed varies, the system dynamics change. This change in system states must be reflected by adapting the weights of the neural network accordingly. The reference neural network model as described earlier sets the desired controller gains as targets for the adaptive ANN while it tries to adapt the weights to achieve them. This adaptive ANN provides the optimum pitch settings to the wind turbine.

### 3.1. Back Propagation Neural Network

Figure 4 shows the basic configuration of a back propagation based neural network with L layers and with \( x_p \) as the
that sum of square error given by (6) will be minimized.

\[ E(W) = \frac{1}{2} \sum_{k=1}^{N_L} [z_{lk}(x) - T_k(x)]^2 \]  

(6)

where, \( E \) is the sum of square error of all the weights \( W \), \( x \) is the input vector, \( T_k \) is the set of target vector and \( N_L \) is the number of neurons in layer \( L \) and \( z_{lk} \) represents the output of the neuron given by (7).

\[ z_{lk} = \left( \sum_{j=1}^{N_i-1} w_{lj}z_{l-1,j} \right) f \]  

(7)

The output of the neuron is transferred to another layer using an activation function, which is a sigmoid function in this case given by (8).

\[ f(t) = \frac{1}{1 + e^{-\alpha t}} \]  

(8)

The updating of weights for the neural network takes place at each time step and is represented by (28).

\[ w_{lj}(t+1) = w_{lj}(t) - \mu \frac{\partial E(W)}{\partial w_{lj}} |_{W(t)} \]  

\[ = w_{lj}(t) - \mu \sum_{p=1}^{n} \frac{\partial E(W)}{\partial w_{lj}} |_{W(t)} \]  

(9)

where \( \mu \) is a learning parameter used for tuning the speed and quality of the learning process. The neural network is said to be trained once the change in error, in weights is minimized given by \( \partial E(W)/\partial w_{lj} \).

### 3.2. Differential Evolution

The DE is a evolutionary search algorithm which finds the optimum value of an objective function subject to satisfying the system constraints \([5][7]\). This makes use of crossover and mutation factors that allow to search for the global optimum in the entire search environment between the upper and lower bounds of the control variables that need to be optimized; in this case the controller parameters \( K_p \) and \( K_i \). In order to choose between the new members that are produced during the mutation and crossover stages, there is a process of survival of the fittest based on the objective function. The objective function, \( f \), chosen for this particular problem is to minimize an eigenvalue based function given by,

\[ f = \sum_{i=1}^{n} (\zeta - \zeta_0) \]  

(10)

where \( n \) represents the eigenvalues of the linearized system of the original non-linear system, \( \zeta \) is the damping ratio of a particular eigenvalue and \( \zeta_0 \) is the predefined damping which has to be achieved during optimization.

The steps involved in the algorithm are briefly given in the following \([5][8]\).

a. Define the dimension of the problem. Since in this case the control variables are, \( K_p \) and \( K_i \), dimension is 2. Also, set maximum and minimum range of the variables.

b. Within the upper and lower bounds for \( K_p \) and \( K_i \), create the population members through the relation,

\[ x_{i,j} = x_{j,min} + rand(0,1) \]  

\[ (x_{j,max} - x_{j,min}) \]  

\[ i = 1, NP; j = 1, D \]  

(11)

c. To change each member of the target generation \( X_i^{(G)} \), a donor vector \( V_i^{(G+1)} \) is produced given by mutation as,

\[ V_i^{(G+1)} = \chi_{r1}^{(G)} + f \left( \chi_{r2}^{(G)} - \chi_{r3}^{(G)} \right) \]  

(12)

\( \chi_{r1}^{(G)} \), \( \chi_{r2}^{(G)} \) and \( \chi_{r3}^{(G)} \) are randomly selected solution vectors from the target generation, \( f \) is the mutation factor.

d. Crossover is applied on each variable defined by,

\[ u_{i,j}^{(G)} \]  

\[ = \begin{cases} v_{i,j}^{(G)} & \text{if } rand(0,1) < CR, \\ x_{i,j}^{(G)} & \text{otherwise} \end{cases} \]  

(13)

\( CR \) is the crossover factor, \( u_{i,j}^{(G)} \), \( v_{i,j}^{(G)} \) and \( x_{i,j}^{(G)} \) is the jth component of the trail vector, donor vector and target vector respectively in the ith population members.

e. To keep the generation size constant select which is going to survive in the next generation by Survival of the Fittest concept using,

\[ X_i^{(G+1)} \]  

\[ = \begin{cases} V_i^{(G)} & \text{if } f \left( V_i^{(G)} \right) \leq f \left( X_i^{(G)} \right), \\ X_i^{(G)} & \text{if } f \left( V_i^{(G)} \right) > f \left( X_i^{(G)} \right) \end{cases} \]  

(14)
Figure 2. Adaptive Neural Network based pitch controller

\( U_i^{(G)} \) is the current trial vector and \( X_i^{(G)} \) is the current target vector. \( J \) is the objective function to be minimized given by,

\[
J = \sum_{i=1}^{n} (\zeta - \zeta_0)
\]

(15)

where \( n \) represents the eigenvalues of the linearized system of the original non-linear system, \( \zeta \) is the damping ratio of a particular eigenvalue and \( \zeta_0 \) is the predefined damping which has to be achieved.

f. The best solution corresponds to the minimum value of the objective function. Repeat steps ‘a-e’, to get the global best values of \( K_p \) and \( K_i \), or stop when the maximum number of iterations has been reached and restart the procedure.

4. ADAPTIVE NEURAL NETWORK

Figure 2 shows the adaptive pitch control strategy employed for DFIG. The strategy involves making use of input wind speed and setting the reference power by making use of Fig. ??.. The reference neural network model uses a back propagation based neural network which sets the PI gains as targets with random wind speeds as inputs. Using differential evolution, the training data for this neural network is obtained. The PI gains obtained from DE are optimal gains by making use of eigenvalue based objective function.

As the wind speed varies, the system dynamics change. This change in system states must be reflected by adapting the weights of the neural network accordingly. The reference neural network model as described earlier sets the desired controller gains as targets for the adaptive ANN while it tries to adapt the weights to achieve them. This adaptive ANN provides the optimum pitch settings to the wind turbine.

Since the back propagation based neural network takes wind speeds as the only input with targets set as controller gains, an adaptive neural network is developed that caters for variation in system states including the changes in wind speeds. For this the weights of the neural network need to be updated at each time step in order to achieve the desired targets set by the reference neural network model. The adaptive neural network makes use of \( \mu \)-LMS (Least Mean Square) algorithm [9]. This algorithm works by performing approximate steepest descent on the weights. An instantaneous gradient, based upon the square of the instantaneous linear error is defined by (16).

\[
\hat{E}_t = \frac{\partial \varepsilon_i^2}{\partial W(t)} = \begin{bmatrix}
\frac{\partial \varepsilon_i^2}{\partial w_{11}} \\
\frac{\partial \varepsilon_i^2}{\partial w_{21}} \\
\vdots \\
\frac{\partial \varepsilon_i^2}{\partial w_{N1}}
\end{bmatrix}
\]

(16)

In (16), \( \varepsilon_i \) holds the instantaneous errors for the neural network, while \( \hat{E}_t \) is a matrix of gradients of errors with respect to the respective weights \( w_0 \ldots w_N \) in the weight matrix \( W \). The weight update is carried out by,

\[
W(t + 1) = W(t) - \mu \hat{E}_t
\]

(17)

where, the weight matrix is updated at each time instant \( t \), with a given learning parameter \( \mu \). The instantaneous gradients are available from the training data. Computing these would involve averaging the instantaneous gradient associated with all patterns in the training set. Applying differentiation on (17) gives,

\[
W(t + 1) = W(t) - 2\mu \varepsilon_i \frac{\partial \varepsilon_i^2}{\partial W(t)}
\]

(18)

\( \varepsilon_i \) gives the linear difference between the desired response at any stage \( d_i \) and the output \( W_i^T x_i \). Replacing \( \varepsilon_i \) in (18) results in.

\[
W(t + 1) = W(t) - 2\mu \varepsilon_i \frac{\partial (d_i - W_i^T x_i)}{\partial W_i}
\]

(19)

Noting that \( d_i \) and \( x_i \) are independent of the current instant of weight value \( W_i \), (19) becomes:

\[
W(t + 1) = W(t) + 2\mu \varepsilon_i x_i
\]

(20)

The learning constant \( \mu \) determines the stability and the convergence rate as discussed in the previous section. In this algorithm, and other iterative steepest descent procedures, use of the instantaneous gradient is perfectly justified if the step size is small.

5. SIMULATION RESULTS

For simulation purposes, it is assumed that initially the system is operating at 12 m/s wind speed, delivering electrical
power at 0.90 p.u. The wind gust, as shown in figure 6, is applied at time t=1sec. Simulations are carried out and the results of a pitched controlled system are compared with a nominal control (KP=1 and KI=0). The wind data, for training the back propagation neural network as described in section A, was collected from the 5-kW wind system installed at the King Fahd University beach front. The simulator used for this paper is MATLAB.

Figure 7 shows the variation in mechanical and electrical power. As the pitch angle adjusts to sudden changes in wind speeds as illustrated in figure 8, the mechanical power also varies accordingly. From the electrical power plot, it is clear that the output power tracks the mechanical power that is available from the wind. It is also observed that contrary to the rapid changes in mechanical power, the electrical power supplied by the wind generator is smooth with minimum transients with an adaptive neural network based pitch controller. The pitch angle variation can be compared with the wind speed data. The primary purpose of the pitch controller is to reduce the pitch angle when the wind speed goes below the rated wind speed and vice versa. Comparing the results of figure 7 with figure 8, it is concluded that the pitch controller is performing the same task.

A controlled response of the generator speed and terminal voltage with the adaptive neural network based pitch controller is also observed from figures 9-10. The transients are reduced significantly as compared to the nominal control as these states change under the given wind conditions. The controller parameters are updated regularly with changes in wind speeds and system states. This is clear from figures 11 and 12. This continuous update of gains allow for increased damping of the states. The nominal control is unable to cope with random change in wind speeds and hence results in minimum damping and poor system response as compared to the adaptive neural network based controller. The controller parameters in the case of nominal control are set to their original values throughout the simulation time.
6. CONCLUSION

The paper presented a complete dynamical model of DFIG system equipped with converter circuitry and pitch control. The pitch controller has been designed using adaptive neural network, making use of back propagation based neural network and differential evolution. The differential evolution provided the training data composed of optimal controller gains as targets for random wind speeds as inputs. The adaptive neural network updates its weights as the system states vary with changes in wind speeds and generates the necessary control. An examination of the results show the effectiveness of the proposed control strategy over the nominal control, as the electrical power output in the case of adaptive neural network tracks the mechanical power, available from wind, faithfully with minimum transients. Other system states like generator speed, terminal voltage and stator current also undergo a controlled response under varying wind speed conditions. The controller gains also adapt themselves with variation in system states to provide the necessary damping.

7. ACKNOWLEDGEMENT

The work was done as part of KFUPM research group projects RG 1202-1 & 1202-2. The authors wish to acknowledge the support of the King Fahd University of Petroleum and Minerals and Prince Mohammad Bin Fahd University.

Biography

If space permits, include a brief biography of no more than 300 words for each author at the end of the article to give it greater impact and validity for the audience.

REFERENCES


